1，下面哪些语句可以在JS里判断一个对象oStringObject是否为String。

* oStringObject instanceof String
* typeof oStringObject == 'string'

JS里面string初始化有两种方式：直接赋值和String对象的实例化。

var str =’abc’;

var str =new String(‘abc’);

通常来说，判断一个对象的类型使用typeof。但是在new String的情况下结果会是object。

此时判断应通过instanceof。

**vtr instanceof String.**

2，常见的浏览器端存储技术有哪些。cookie localStorage userData

**浏览器端：**

cookie

WebStorage(localStorage、sessionStorage)

userData

indexedDB

**服务器端：**

session

有时需要将网页中的一些数据保存在浏览器端，这样做的好处是，当下次访问页面时，不需要再次向服务器请求数据，直接就可以从本地读取数据。目前常用的有以下几种方法：

**cookie**

cookie会随着每次HTTP请求头信息一起发送，无形中增加了网络流量，另外，cookie能存储的数据容量有限，根据浏览器类型不同而不同，IE6大约只能存储2K。

**Flash ShareObject**

这种方式能能解决上面提到的cookie存储的两个弊端，而且能够跨浏览器，应该说是目前最好的本地存储方案。不过，需要在页面中插入一个Flash，当浏览器没有安装Flash控件时就不能用了。所幸的是，没有安装Flash的用户极少。

缺点：需要安装Flash插件。

**Google Gear**

Google开发出的一种本地存储技术。

缺点：需要安装Gear组件。

**userData**

IE浏览器可以使用userData来存储数据，容量可达到640K，这种方案是很可靠的，不需要安装额外的插件。缺点：它仅在IE下有效。

**sessionStorage**

使用于Firefox2+的火狐浏览器，用这种方式存储的数据仅窗口级别有效，同一个窗口（或者Tab）页面刷新或者跳转，都能获取到本地存储的数据，当新开窗口或者页面时，原来的数据就失效了。

缺点：IE不支持、不能实现数据的持久保存。

**globalStorage**

使用于Firefox2+的火狐浏览器，类似于IE的userData。

3.JS中的基本数据类型

1），undefined。2），boolean。3），Null。4)，string。5），Number。